

Narration	Time
Loops in awk پراسپوکن ٹیوٹوریل میں خوش آمدید۔	00:01
اس ٹیوٹوریل میں ہم	00:06
while, do-while, for اور مزید awk میں looping constructs کے بارے میں سیکھیں گے	
ہم کچھ مثالوں کے ذریعے سیکھیں گے۔	00:16
اس ٹیوٹوریل کو ریکارڈ کرنے کے لئے میں	00:20
Ubuntu لینکس 16.04 آپریٹنگ سسٹم اور gedit text editor 3.20.1 استعمال کر رہا ہوں	
آپ اپنی پسند کا کوئی بھی ٹیکسٹ ایڈیٹر استعمال کر سکتے ہیں۔	00:32
اس ٹیوٹوریل پر عمل کرنے کے لئے، آپ کو ہماری ویب سائٹ پر پچھلے AWK ٹیوٹوریل پڑھنے چاہئے۔	00:36
آپ کو کسی بھی پروگرامنگ لینگویج جیسے C یا ++C سے بھی واقف ہونا چاہئے۔	00:43
اگر نہیں، تو ہماری ویب سائٹ پر متعلقہ مضامین کا مطالعہ کریں	00:50
اس ٹیوٹوریل میں استعمال کردہ فائلیں اس ٹیوٹوریل پیج پر کوڈ فائلوں میں موجود ہیں۔ ان کو ڈاؤن لوڈ کریں اور extract کریں	00:56
لوپ ہمیں ایک یا زیادہ آپریشنز کو بار بار کرنے کی اجازت دیتا ہے۔	01:06
while, do-while اور for، awk میں دستیاب loops ہیں	01:12
while' loop کے سٹیکس کو یہاں دیکھا جاسکتا ہے۔	01:18
While loop چیک کرتا ہے کہ کیا مخصوص condition، true ہے	01:22
اگر ہاں، تو یہ code کو body کے اندر execute کرتا ہے۔ یہ لوپ while تب تک دہرایا جائے گا جب تک متعلقہ کنڈیشن کے true ہے	01:27
ہم اسی awkdemo.txt فائل کا استعمال کریں گے، جس سے ہم نے پہلے استعمال کیا تھا	01:37
میں نے پہلے ہی while_loop.awk نامی script لکھا ہے۔	01:43
اس فائل میں کوڈ فائلوں کے لنکس میں دستیاب ہے۔	01:48
یہاں ہم نے field separator کو بطور Pipe علامت متعین کیا ہے	01:53
ابتدائی طور پر، ہمیں i loop variable کی ویلیو 1 مقرر کرنے کی ضرورت ہے۔	01:58

یہاں ہمارے پاس ایک اور ریبل $f$ ہے، جو 1 کے ساتھ انشلائز ہوا ہے	02:04
Variable $f$ ہر ریکارڈ کے لئے field counter یا fields کی پوزیشن کی نمائندگی کرتا ہے۔	02:10
اب while condition میں ہم چیک کریں کہ آیا $i$ کی ویلیو 3 سے کم یا برابر ہے۔	02:17
اگر ہاں، تو <code>awkdemo.txt</code> فائل میں اس ریکارڈ کیلئے <code>fth field</code> 'پرنت کریں گے۔	02:23
پھر ہم <code>f</code> field counter میں 1 کا اضافہ کریں گے۔	02:31
اس کے بعد، ہم <code>loop variable</code> کی ویلیو میں بھی 1 کا اضافہ کریں گے۔	02:36
یہ <code>printf</code> ہر row کے اختتام پر <code>newline character</code> پرنت کرتا ہے۔	02:43
یہ <code>loop</code> ، <code>awkdemo.txt</code> فائل میں تمام ریکارڈز کے لئے ایکز کیوٹ کی جائے گی۔	02:49
اس کا مطلب یہ ہے کہ ہر ریکارڈ کیلئے پہلے 3 فیلڈس کو پرنت کیا جائے گا۔	02:55
اب اس کوڈ کو ایکز کیوٹ کریں	03:00
<code>Ctrl</code> ، <code>Alt</code> اور <code>T</code> کیز دبا کر ٹرمینل کھولیں۔	03:03
<code>cd command</code> کمانڈ کا استعمال کرتے ہوئے، اس فولڈر میں جائیں جہاں آپ نے کوڈ فائلوں کو ڈاؤن لوڈ اور ایکسٹریکٹ کیا ہے۔	03:09
اب ٹائپ کریں: <code>awk space hyphen small f space while_loop.awk space</code> <code>awkdemo.txt</code> دبا <code>Enter</code> ۔	03:16
نوٹ کریں کہ ہمیں آؤٹ پٹ میں ہر ریکارڈ کے 3 فیلڈس ملے ہیں	03:29
<code>do-while loop</code> کے ساتھ بھی ایسا ہی کرتے ہیں	03:35
یہاں <code>do-while loop</code> کا سٹیکس دیکھا جاسکتا ہے۔	03:38
<code>do-while loop</code> ہمیشہ <code>body</code> میں ایک بار ہی کوڈ کو انجام دیتا ہے۔	03:42
پھر یہ مخصوص <code>condition</code> کی جانچ کرتا ہے۔ اور جب تک مخصوص <code>condition</code> ، <code>true</code> ہے، کوڈ <code>body</code> کے اندر دہرایا جاتا ہے	03:47
میں نے پہلے ہی ایک سکرپٹ لکھا ہے اور اسے <code>do_loop.awk</code> نام دیا ہے یہی فائل <code>Code Files</code> کے لنک میں دستیاب ہے۔	03:56

04:06	اس کوڈ میں، یہ do loop کے اندر کے وہ statements ہیں جنہیں پہلے ایکزیکوٹ کیا جائے گا۔ یہ وہ condition ہے جسے جانچا جائے گا
04:15	اس کے بعد، loop میں statements تب تک دہرائے جائیں گے جب تک condition، True ہوگی
04:23	یہ لوپ awkdemo.txt فائل میں تمام ریکارڈوں کے لئے دہرایا جائے گا، جس کا مطلب ہے کہ سب سے پہلے 3 فیلڈس کو تمام ریکارڈز کے لئے پرنٹ کیا جائے گا۔
04:33	ٹرینل پر جائیں ٹرینل صاف کریں۔
04:38	اب ٹائپ کریں: awk space hyphen small f space do underscore loop dot awk space awkdemo dot txt Enter دبائیں۔
04:52	ہم وہی آؤٹ پٹ حاصل کرتے ہیں۔ پھر ہمیں while اور do-while loops کی کیا ضرورت
04:58	فرق سمجھیں۔
05:00	while underscore loop dot awk فائل پر جائیں۔
05:05	اب، loop counter کی ویلیو 1 بنائیں
05:11	یہ ابتداء سے مخصوص کنڈیشن کو false بنائے گا، مطلب ہمیں کوئی آؤٹ پٹ نہیں ملنا چاہئے۔
05:19	فائل Save کریں اور terminal پر جائیں۔
05:22	terminal صاف کریں۔ اب جب تک آپ کو while loop ایکزیکوٹ کرنے کی کمانڈ حاصل نہ ہو، up-arrow کی دباتے رہیں
05:30	اب اینٹر دبائیں۔
05:32	دیکھیں، ہم خالی لائنوں کے علاوہ کوئی بھی آؤٹ پٹ نہیں ملتا
05:37	awkdemo.txt فائل میں ہر ریکارڈ کے لئے آؤٹ پٹ میں خالی لائنوں کو پرنٹ کیا جا رہا ہے۔
05:44	اب، do loop فائل میں کچھ تبدیلیاں کرتے ہیں۔
05:48	فائل پر جائیں do underscore loop dot awk
05:53	i کی ویلیو 1 سے 4 تک بڑھائیں
05:57	فائل کو سیو کریں اور ٹرینل پر جائیں۔

06:01	ٹرمینل صاف کریں۔ جب تک آپ کو do loop ایکڑ کیوٹ کرنے کی کمانڈ نہ حاصل ہو، up-arrow کی دباتے رہیں
06:10	آؤٹ پٹ میں، ہر رو کیلئے پہلا فیلڈ پرنٹ کیا جاتا ہے۔ وجہ کیا ہے
06:16	ہر رو کے لئے، awk سب سے پہلے، پہلے فیلڈ میں ویلیو پرنٹ کرتا ہے، کیونکہ variable f ، 1 پر انشلاز کیا جاتا ہے۔ اور پھر اس کی جانچ کی جاتی ہے۔
06:28	چونکہ loop counter i کی ویلیو 4 ہے، تو condition ، false ہے۔ تو اس ریکارڈ کے لئے لوپ یہیں ختم ہو جاتا ہے۔
06:39	یہ loop، awkdemo.txt فائل میں تمام ریکارڈوں کے لئے دہرایا جائے گا۔
06:44	جس کا مطلب ہے ہر Record کے لئے پہلا field پرنٹ کیا جائے گا۔
06:49	ہم کم از کم ایک بار ہر ریکارڈ کے لئے آؤٹ پٹ حاصل کر رہے ہیں۔
06:53	کسی بھی دوسری condition سے بے نیاز ہو کر ایکڑ کیوٹ کئے جانے والے کام کیلئے، do-while loop کا استعمال کریں۔
07:01	ہم یہ for loop کے ساتھ بھی کر سکتے ہیں۔
07:05	for loop کے لئے سٹیکس یہاں دیکھا جاسکتا ہے۔
07:09	'for' statement ، initialization کو ایکڑ کیوٹ کر کے شروع ہوتا ہے۔
07:14	جب تک condition ، true ہے، تب تک یہ اندر کے statements بار بار ایکڑ کیوٹ کرتا رہتا ہے اور پھر increments کرتا ہے
07:23	یہ مانتے ہوئے کہ آپ کو C یا ++C لینگویج کی علمیت ہے، میں تفصیل میں وضاحت نہیں کر رہا ہوں۔
07:30	اس طرح اس condition کیلئے for loop ایسا لگتا ہے
07:35	یہاں initialization, condition کی جانچ اور ویریبل ایک ہی لائن میں کئے جاتے ہیں
07:43	یہ مشق اپنے آپ کریں۔
07:46	یہاں کچھ اور looping constructs ہیں۔ break, continue, exit.
07:53	ہم بعد میں ان کے بارے میں متعلقہ مثالیں دیکھیں گے۔

07:58	ہماری فائل میں سنگل اور ملٹی لائن کمیٹس ہو سکتے ہیں۔
08:03	یہاں نوٹ کریں کہ سنگل لائن کمیٹس (#) hash 'علامت کے ساتھ ڈکلیئر کی جاتی ہے
08:10	ڈبل hash (##) علامت کی مدد سے ملٹی لائن ڈکلیئر کئے جاتے ہیں
08:16	اب، آؤٹ پٹ میں ان comments کی جانچ اور پرنٹنگ کا کوئی مطلب نہیں ہے۔
08:22	ہمیں hash (##) کی علامت پر شروع ہونے والی لائن چھوڑنا ہے۔ ہم یہ کیسے کر سکتے ہیں؟
08:28	یاد کریں ان لوگوں کو جن کی 8000 روپے سے زائد آمدن ہونے پر 50% اضافہ کیا گیا تھا
08:36	ہم comments چھوڑنے کے لئے اسی مثال کا استعمال کریں گے۔
08:40	جیسا کہ یہاں دکھایا گیا ہے، میں نے یہاں next.awk نامی فائل ایکزیکوشن کیلئے بنائی
08:47	اب، اس command کا کیا مطلب ہے؟
08:50	awk، ہر لائن کے ابتدائین caret sign hash symbol (^#) تلاش کرے گا
08:57	اگر پیٹرن پایا جاتا ہے، تو کی وارڈ next، awk سے موجودہ لائن فوری طور پر چھوڑنے کو کہتا ہے
09:04	پھر AWK، فائل کی اگلی لائن سے پروسیسنگ شروع کرے گا۔ یہ پروسیسنگ کا وقت بچائے گا۔
09:12	ٹرینل پر جائیں اور دکھائی گئی کمانڈ ٹائپ کریں۔ اور اینٹر دبائیں۔
09:20	ہم comments کے بغیر آؤٹ پٹ حاصل کرتے ہیں۔
09:24	فرض کریں، ہمارے پاس طالب علموں کے ریکارڈوں کی ایک ہی فارمیٹ میں بہت سے فائلیں ہیں۔ مثلاً awkdemo2.txt اور awkdemo_mod.txt
09:37	دیکھیں، یہ ہماری کچھلی فائل کی طرح ہے۔
09:41	اس کے سامنے ہیش کی علامت کے ساتھ comments ہیں۔
09:45	اور آخر میں ڈبل ہیش ## علامت کے ساتھ بہت سا ٹیکسٹ ہے۔
09:50	لہذا ہمارا ڈیٹا دو مختلف فائلوں میں ہے۔ تو awk کو تمام طالب علموں کو increment دینے کیلئے دونوں فائلوں پر عملدرآمد کرنا چاہئے۔
09:59	جب ہم پہلی ہی فائل کے ڈبل hash (##) کے نشان پر پہنچ جاتے ہیں تو، awk اس فائل کی پروسیسنگ کو مکمل طور پر روکنا چاہئے۔

10:06	پھر اگلی فائل سے ایکڑ، کیوشن شروع کرنا ہوگی۔ یہ پروسسنگ کے وقت کو بچائے گا۔
10:13	اگلے کوڈ میں ترمیم کریں جیسا کہ یہاں دکھایا گیا ہے۔
10:17	میں نے BEGIN اسٹیٹمنٹ کے نیچے dollar zero tilde slash caret symbol double hash slash within braces nextfile semicolon شامل کیا ہے۔
10:29	یہ ہر لائن کے شروع میں یہ # hash علامت تلاش کرے گا
10:34	اگر پائی جاتی ہے، تو پھر اگلی فائل پر عمل کرنے کیلئے awk موجودہ فائل کو چھوڑ دیتا ہے۔
10:39	اس فائل کو سیو کریں۔
10:41	ٹرینل پر جائیں اور مندرجہ ذیل کمانڈ ٹائپ کریں۔ Enter دبائیں۔
10:48	دیکھیں، ہم دونوں فائلوں سے آؤٹ پٹ حاصل کر رہے ہیں۔
10:53	اس کے ساتھ ہم اس ٹیوٹوریل کے اختتام پر آگئے گئے ہیں۔ مختصر میں
10:58	اس ٹیوٹوریل میں ہم نے AWK میں while, do... while, for, next, nextfile سیکھا
11:06	مشق کے طور پر، awkdemo2.txt کے طلباء کے رکارڈس کیلئے صرف جفت فیلڈ پرنٹ کریں (مثلاً فیلڈ 2، فیلڈ 4 وغیرہ) اس سے قطع نظر کہ input file میں کتنے رکارڈس ہیں
11:22	مندرجہ ذیل لنک پر دستیاب ویڈیو اسپوکن ٹیوٹوریل پروجیکٹ کا خلاصہ کرتا ہے۔ آپ اسے ڈاؤن لوڈ کر کے دیکھ سکتے ہیں۔
11:30	اسپوکن ٹیوٹوریل پروجیکٹ ٹیم اسپوکن ٹیوٹوریلز کا استعمال کرتے ہوئے ورکشاپس منعقد کرتی ہے اور آن لائن امتحان پاس کرنے والوں کو اسناد دیتی ہے۔ مزید معلومات کے لئے، ہمیں لکھیں۔
11:43	کیا آپ اس ٹیوٹوریل کے بارے میں کوئی سوالات رکھتے ہیں؟ براہ ہربانی یہ سائٹ ملاحظہ کریں۔
11:49	اسپوکن ٹیوٹوریل پروجیکٹ کو MHRD، بھارتی حکومت کے قومی مشن برائے تعلیم نے ICT کے ذریعے معاونت کی ہے۔ اس مشن پر مزید معلومات مندرجہ ذیل لنک پر دستیاب ہیں
12:01	آئی آئی ٹی بمبئی سے میں وجاحت احمد اب آپ سے اجازت لیتا ہوں، ہمارے ساتھ شامل ہونے کا شکریہ۔